Filtering with Abstract Particles

Jacob Steinhardt   Percy Liang

Stanford University

{*jsteinhardt,pliang*}*@cs.stanford.edu*

May 1, 2013

Motivation

Goal. Given an (un-normalized) target distribution $f^*(x)$, $p^*(x) = \frac{1}{Z} f^*(x)$, want to compute normalization constant Z.
Issue. Often computationally intractable, so use some approximation $\hat{f}$ to f^*.

- variational Bayes, expectation propagation (drop dependencies)
- MCMC, sequential Monte Carlo, beam search (use samples)

We will show how to combine advantages of both types of methods.

Variational vs. Particle Methods

Goal: infer missing characters in `r e _ _ _ c e`

Variational vs. Particle Methods

Goal: infer missing characters in `r e _ _ _ c e`

Particle			
0.5	replace		
0.5	retrace		

Variational vs. Particle Methods

Goal: infer missing characters in `r e _ _ _ c e`

Particle		Actual		
0.5	replace	0.33	replace	
0.5	retrace	0.33	retrace	
		0.33	rejoice	
		0.01	...	

Variational vs. Particle Methods

Goal: infer missing characters in `r e _ _ _ c e`

Particle		Actual	
0.5	replace	0.33	replace
0.5	retrace	0.33	retrace
		0.33	rejoice
		0.01	...

Variational

r e	0.33 j	0.33 l	0.66 a	c e
	0.33 p	0.33 o	0.33 i	
	0.33 t	0.33 r		

Variational vs. Particle Methods

Goal: infer missing characters in `r e _ _ _ c e`

Particle		Actual		Variational
0.5	replace	0.33	replace	r e [0.33 j, 0.33 p, 0.33 t] [0.33 l, 0.33 o, 0.33 r] [0.66 a, 0.33 i] c e
0.5	retrace	0.33	retrace	
		0.33	rejoice	
		0.01	...	

Particles provide **precision** but lack **coverage**, while variational inference lacks precision.

Our Proposal

Define approximations over intermediate **regions**.

variational *re⋆⋆⋆ce*

particle *replace* *retrace*

Our Proposal

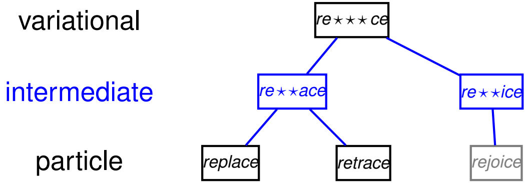
Define approximations over intermediate **regions**.

variational   *re⋆⋆⋆ce*

particle   *replace*   *retrace*   *rejoice*

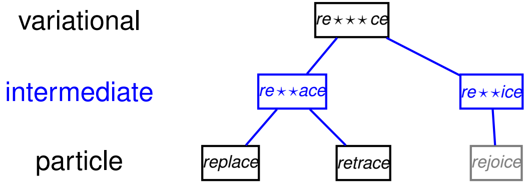
Our Proposal

Define approximations over intermediate **regions**.



Our Proposal

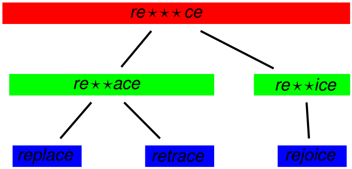
Define approximations over intermediate **regions**.



Goal. Stitch together approximations at multiple levels to simultaneously obtain **precision** (from lower levels) and **coverage** (from higher levels).

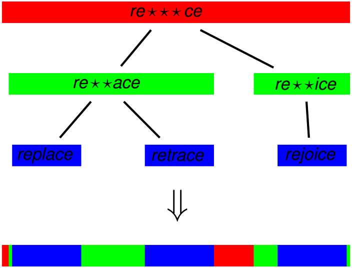
Stitching Together Models

Question. How to combine the different models?



Stitching Together Models

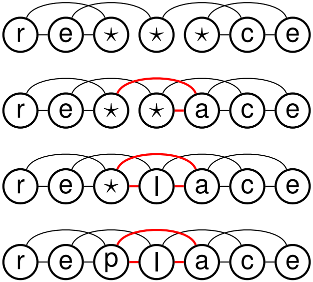
Question. How to combine the different models?



Answer. Just use most precise model available at each point (relies on nested structure, e.g. the regions form a ***hierarchical decomposition***).

Generalizing the Construction

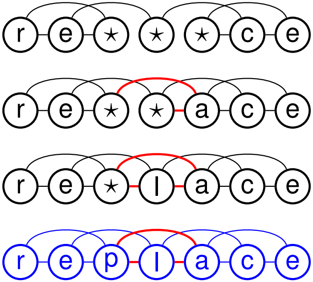
Let X be some space. Suppose we have a hierarchical decomposition $A \subseteq 2^X$ together with an approximation $\hat{f}_a$ to f^* defined on each region $a \in A$.



Generalizing the Construction

Let X be some space. Suppose we have a hierarchical decomposition $A \subseteq 2^X$ together with an approximation $\hat{f}_a$ to f^* defined on each region $a \in A$.

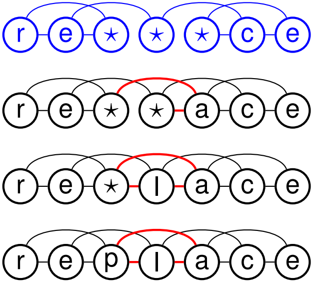
- If $a = \{x_0\}$ is a singleton set, can have $\hat{f}_a(x_0) = f^*(x_0)$.



Generalizing the Construction

Let X be some space. Suppose we have a hierarchical decomposition $A \subseteq 2^X$ together with an approximation $\hat{f}_a$ to f^* defined on each region $a \in A$.

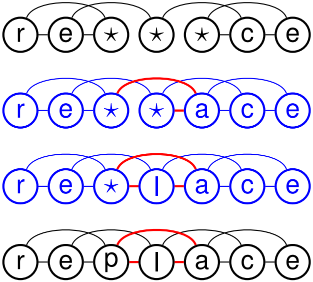
- If $a = \{x_0\}$ is a singleton set, can have $\hat{f}_a(x_0) = f^*(x_0)$.
- If $a = X$, will need to drop most of the dependencies.



Generalizing the Construction

Let X be some space. Suppose we have a hierarchical decomposition $A \subseteq 2^X$ together with an approximation $\hat{f}_a$ to f^* defined on each region $a \in A$.

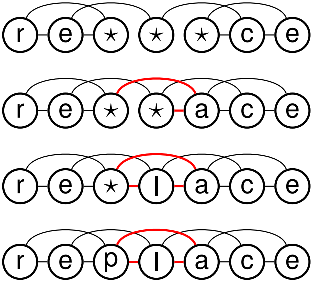
- If $a = \{x_0\}$ is a singleton set, can have $\hat{f}_a(x_0) = f^*(x_0)$.
- If $a = X$, will need to drop most of the dependencies.
- For intermediate values of a (for instance, fixing the values of certain variables) can keep some subset of the dependencies.



Generalizing the Construction

Let X be some space. Suppose we have a hierarchical decomposition $A \subseteq 2^X$ together with an approximation $\hat{f}_a$ to f^* defined on each region $a \in A$.

- If $a = \{x_0\}$ is a singleton set, can have $\hat{f}_a(x_0) = f^*(x_0)$.
- If $a = X$, will need to drop most of the dependencies.
- For intermediate values of a (for instance, fixing the values of certain variables) can keep some subset of the dependencies.

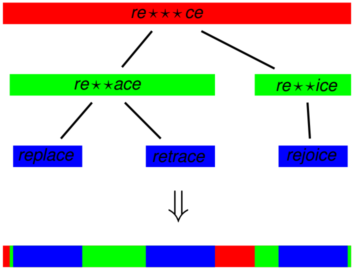


Set $\hat{f}(x) \stackrel{\text{def}}{=} \hat{f}_a(x)$, where a is the smallest region containing x.
Can think of each region $a \in A$ as an ***abstract particle***.

Inference

If $\hat{f}$ is constructed as in the previous slide, then we can compute normalization constant Z as long as we can compute $\sum_{x \in b} \hat{f}_a(x)$ for all regions $b \subseteq a$.

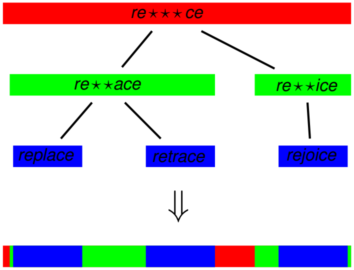
Proof by picture:



Inference

If $\hat{f}$ is constructed as in the previous slide, then we can compute normalization constant Z as long as we can compute $\sum_{x \in b} \hat{f}_a(x)$ for all regions $b \subseteq a$.

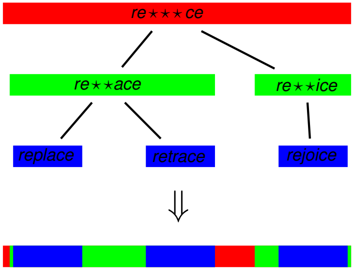
Proof by picture:



Inference

If $\hat{f}$ is constructed as in the previous slide, then we can compute normalization constant Z as long as we can compute $\sum_{x \in b} \hat{f}_a(x)$ for all regions $b \subseteq a$.

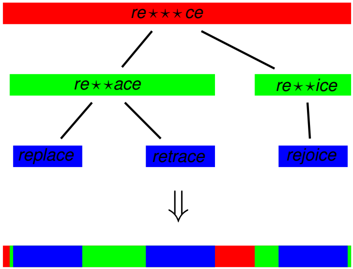
Proof by picture:

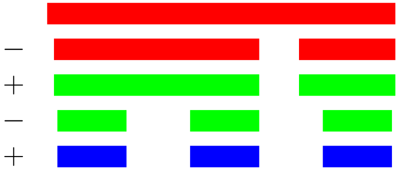


Inference

If $\hat{f}$ is constructed as in the previous slide, then we can compute normalization constant Z as long as we can compute $\sum_{x \in b} \hat{f}_a(x)$ for all regions $b \subseteq a$.

Proof by picture:

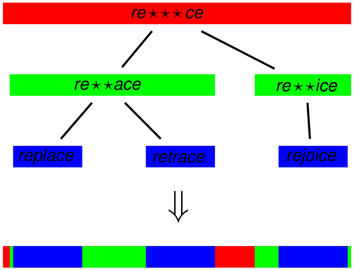


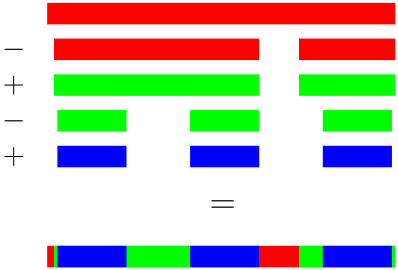


Inference

If $\hat{f}$ is constructed as in the previous slide, then we can compute normalization constant Z as long as we can compute $\sum_{x \in b} \hat{f}_a(x)$ for all regions $b \subseteq a$.

Proof by picture:





A Family of Approximations

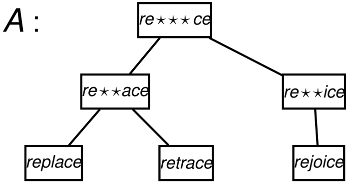
A hierarchical decomposition A leads to an approximation $\hat{f}$.

We would like to define a family of approximations and choose the best one.

Key idea. Every **subset** B of a hierarchical decomposition A is itself a hierarchical decomposition.

- Can let A have **large cardinality** and search for a **small subset** B that yields a good approximation.

Example:



A Family of Approximations

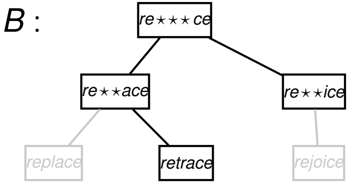
A hierarchical decomposition A leads to an approximation $\hat{f}$.

We would like to define a family of approximations and choose the best one.

Key idea. Every **subset** B of a hierarchical decomposition A is itself a hierarchical decomposition.

- Can let A have **large cardinality** and search for a **small subset** B that yields a good approximation.

Example:



A Family of Approximations

A hierarchical decomposition A leads to an approximation $\hat{f}$.

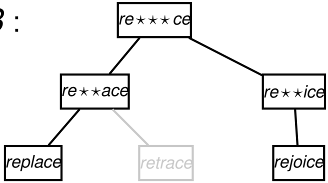
We would like to define a family of approximations and choose the best one.

Key idea. Every **subset** B of a hierarchical decomposition A is itself a hierarchical decomposition.

- Can let A have **large cardinality** and search for a **small subset** B that yields a good approximation.

Example:

B :



A Family of Approximations

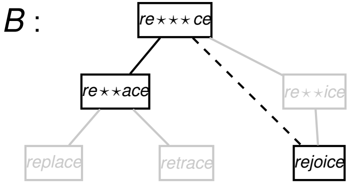
A hierarchical decomposition A leads to an approximation $\hat{f}$.

We would like to define a family of approximations and choose the best one.

Key idea. Every **subset** B of a hierarchical decomposition A is itself a hierarchical decomposition.

- Can let A have **large cardinality** and search for a **small subset** B that yields a good approximation.

Example:



A Family of Approximations

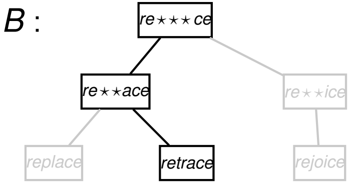
A hierarchical decomposition A leads to an approximation $\hat{f}$.

We would like to define a family of approximations and choose the best one.

Key idea. Every **subset** B of a hierarchical decomposition A is itself a hierarchical decomposition.

- Can let A have **large cardinality** and search for a **small subset** B that yields a good approximation.

Example:



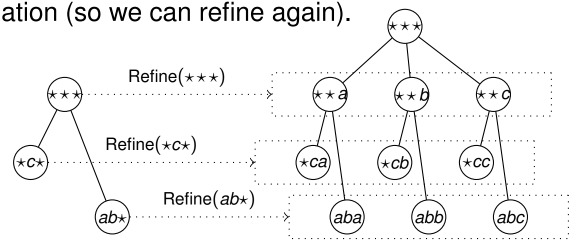
Search Strategy

Suppose that A has size 1000 and we want a subset of size 100.

$\binom{1000}{100}$ possibilities; far too many!

Solution. "Abstract beam search." Iteratively *refine* and *prune* a candidate decomposition.

- **Refine**: split each region into smaller regions (to gain precision).
- **Prune**: greedily keep a small set of regions that yield a good approximation (so we can refine again).



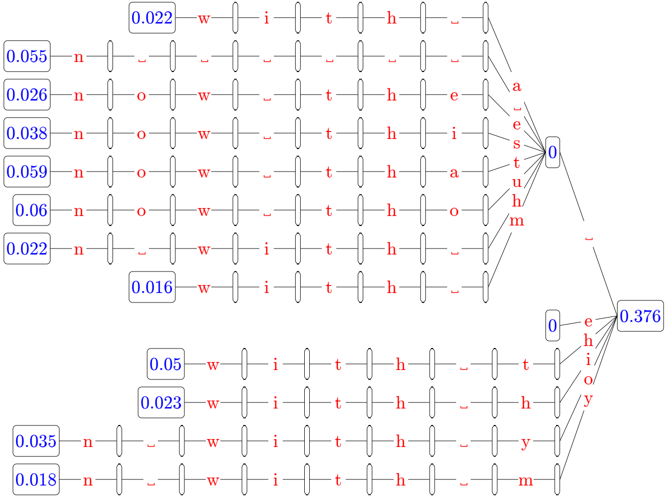
Applies naturally to filtering tasks (refine to go to next time step, prune to save resources).

Summary (so far)

- Interpolate between individual particles and full variational approximations by using region-specific approximations.
- Stitch together approximations in different regions via a hierarchical decomposition.
- Prune and refine the decomposition to find a good approximation.
- Related to split variational inference (Bouchard & Zoeter, 2009).
- Also to a growing family of coarse-to-fine inference methods (Petrov et al., 2006; Weiss & Taskar, 2010; many others).

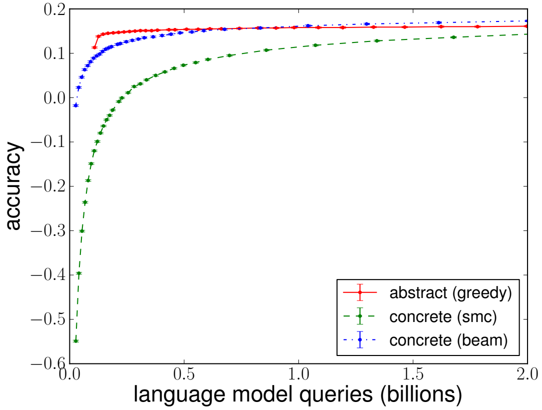
Experiments

Input: ? ? ? n ? w ? t h ? ? ?

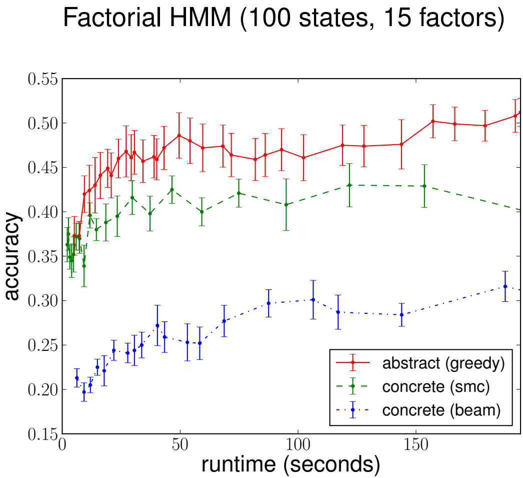


Experiments

n-gram text reconstruction ($n = 8$)



Experiments



Conclusion

- Abstract particles combine the advantages of variational and particle inference.
- Provide a framework for reasoning about the optimal representation for approximate inference.
- Thanks!